

IDENTIFIKASI OBJEK BUAH BERDASARKAN WARNA DENGAN *IMAGE CLUSTERING*

TUGAS AKHIR

Diajukan Sebagai Salah Satu Syarat
Untuk Memperoleh Gelar Sarjana Teknik Pada
Jurusan Teknik Informatika

oleh :

TYAS PARAMITA
10451025571



**FAKULTAS SAINS DAN TEKNOLOGI
UNIVERSITAS ISLAM NEGERI SULTAN SYARIF KASIM RIAU
PEKANBARU
2011**

IDENTIFIKASI OBJEK BUAH BERDASARKAN WARNA DENGAN *IMAGE CLUSTERING*

TYAS PARAMITA
10451025571

Tanggal Sidang : Juni 2011
Tanggal Wisuda : 2011

Jurusan Teknik Informatika
Fakultas Sains dan Teknologi
Universitas Islam Negeri Sultan Syarif Kasim Riau

ABSTRAK

Pengenalan pola digunakan untuk mengenali satu atau lebih objek di dalam sebuah citra. Objek-objek tersebut dapat diklasifikasikan ke dalam kelompok berdasarkan kesamaan tertentu, dinamakan *cluster*.

Dalam tugas akhir ini, dilakukan rancang bangun aplikasi untuk mengidentifikasi objek buah berdasarkan ciri warna (*Red*, *Green* dan *Blue*). *Training* dilakukan untuk mendapatkan nilai centroid data yang akan disimpan ke dalam basis data dengan label nama buah tertentu. Selanjutnya, dilakukan *clustering* menggunakan metode K-Means terhadap ciri warna yang diperoleh. Jumlah *cluster* yang diperoleh menunjukkan jumlah buah yang teridentifikasi. Nama buah diperoleh dengan cara membandingkan nilai centroid tiap *cluster* dengan nilai centroid data *training*.

Berdasarkan hasil pengujian, identifikasi objek buah berdasarkan warna dengan *image clustering* menggunakan metode K-means mempunyai akurasi mencapai 63,5 %.

Kata kunci : ciri warna, *image clustering*, metode K-Means, pengenalan buah

IDENTIFIKASI OBJEK BUAH BERDASARKAN WARNA DENGAN *IMAGE CLUSTERING*

TYAS PARAMITA
10451025571

Tanggal Sidang : Juni 2011
Tanggal Wisuda : 2011

Jurusan Teknik Informatika
Fakultas Sains dan Teknologi
Universitas Islam Negeri Sultan Syarif Kasim Riau

ABSTRACT

Pattern recognition is used to identify one or more objects within an image. Those objects can be classified into group based on specific similarity, called cluster.

In this final project, a computer program was built to identify fruit object by using its colour characteristic (Red, Green and Blue). Centroid value of training set was stored in database and labeled by fruit's name. Then, those features were clustered using K-Means method. Number of cluster represents number of identified fruits. Fruit's name was obtained by comparing between centroid value of each cluster and centroid value of training set.

Based on test result, fruit object identification based on colour using image clustering was 63,5 %.

Keyword : colour characteristic, K-Means method, fruit identification, image clustering.

DAFTAR ISI

Halaman

ABSTRAK.....	Error! Bookmark not defined.
KATA PENGANTAR	Error! Bookmark not defined.
DAFTAR ISI.....	1
DAFTAR GAMBAR	Error! Bookmark not defined.
DAFTAR TABEL.....	Error! Bookmark not defined.
BAB I PENDAHULUAN.....	Error! Bookmark not defined.
1.1 Latar Belakang	Error! Bookmark not defined.
1.2 Rumusan Masalah	Error! Bookmark not defined.
1.3 Batasan Masalah.....	Error! Bookmark not defined.
1.4 Tujuan.....	Error! Bookmark not defined.
1.5 Sistematika Penulisan.....	Error! Bookmark not defined.
BAB II LANDASAN TEORI.....	Error! Bookmark not defined.
2.1 Pengolahan Citra	Error! Bookmark not defined.
2.1.1 Citra Digital	Error! Bookmark not defined.
2.1.2 Jenis Citra	Error! Bookmark not defined.
2.1.3 Format File Citra	Error! Bookmark not defined.
2.2 <i>Clustering</i>	Error! Bookmark not defined.
2.2.1 Karakteristik <i>Clustering</i>	Error! Bookmark not defined.
2.2.2 Algoritma <i>Clustering</i>	Error! Bookmark not defined.
2.2.3 Euclidean Distance	Error! Bookmark not defined.
BAB III METODOLOGI PENELITIAN	Error! Bookmark not defined.
3.1 Penelitian Pendahuluan	Error! Bookmark not defined.
3.2 Identifikasi Masalah	Error! Bookmark not defined.
3.3 Penetapan Tujuan	Error! Bookmark not defined.
3.4 Pengumpulan Data	Error! Bookmark not defined.
3.5 Analisa dan Perancangan	Error! Bookmark not defined.
3.6 Implementasi dan Pengujian	Error! Bookmark not defined.

3.7	Kesimpulan dan Saran.....	Error! Bookmark not defined.
BAB IV ANALISA DAN PERANCANGAN.....		Error! Bookmark not defined.
4.1	Analisa.....	Error! Bookmark not defined.
4.1.1	Deskripsi Umum.....	Error! Bookmark not defined.
4.1.2	Citra Masukan	Error! Bookmark not defined.
4.1.3	Proses Pembelajaran	Error! Bookmark not defined.
4.1.4	Proses Identifikasi	Error! Bookmark not defined.
4.2	Perancangan	Error! Bookmark not defined.
4.2.1	Perancangan Basis Data	Error! Bookmark not defined.
4.2.2	Perancangan Struktur Menu.....	Error! Bookmark not defined.
4.2.3	Perancangan Antar Muka ...	Error! Bookmark not defined.
BAB V IMPLEMENTASI DAN PENGUJIAN		Error! Bookmark not defined.
5.1	Implementasi	Error! Bookmark not defined.
5.1.1	Lingkungan Implementasi ..	Error! Bookmark not defined.
5.1.2	Hasil Implementasi	Error! Bookmark not defined.
5.2	Pengujian	Error! Bookmark not defined.
5.2.1	Proses Pembelajaran	Error! Bookmark not defined.
5.2.2	Analisa Hasil Pembelajaran	Error! Bookmark not defined.
5.2.3	Pengujian Sistem Identifikasi Buah.....	Error! Bookmark not defined.
BAB VI PENUTUP		Error! Bookmark not defined.
6.1	Kesimpulan.....	Error! Bookmark not defined.
6.2	Saran.....	Error! Bookmark not defined.
DAFTAR PUSTAKA		Error! Bookmark not defined.

DAFTAR TABEL

Tabel	Halaman
4. 1 Struktur Tabel Buah	Error! Bookmark not defined.
5. 1 Hasil Pembelajaran	Error! Bookmark not defined.

BAB I

PENDAHULUAN

1.1 Latar Belakang

Pengenalan pola (*pattern recognition/image interpretation*) merupakan salah satu bidang studi yang berkaitan dengan citra (*image*) dalam ilmu komputer. Pengenalan pola mengelompokkan data numerik dan simbolik (termasuk citra) secara otomatis oleh komputer. Tujuan dilakukannya pengelompokan yaitu untuk mengenali satu atau lebih objek di dalam citra. Manusia dapat mengenali objek yang dilihatnya karena otak manusia telah belajar mengklasifikasikan objek-objek yang ada. Hal tersebut memungkinkan manusia untuk membedakan suatu objek dengan objek lainnya. Kemampuan sistem visual manusia inilah yang dicoba ditiru oleh mesin, dimana komputer menerima masukan berupa citra dari objek yang akan diidentifikasi, mengolah citra tersebut, dan memberikan keluaran berupa deskripsi objek di dalam citra.

Dewasa ini banyak bermunculan penelitian – penelitian baru dibidang pengenalan pola. Aplikasi pengenalan pola yang telah ada salah satunya adalah proses identifikasi / pengenalan buah berdasarkan ciri warna. Pada pengenalan buah berdasarkan ciri warna sebelumnya, sistem hanya mampu mengidentifikasi objek buah tunggal. Sistem yang dibangun belum mampu mengenali objek berupa sekumpulan buah. Sehingga dibutuhkan suatu solusi untuk permasalahan tersebut.

Dalam tugas akhir ini dilakukan rancang bangun aplikasi identifikasi objek buah berdasarkan warna dengan *image clustering*. Dimana objek buah yang digunakan adalah sekumpulan buah, bukan buah tunggal. Metode yang digunakan adalah metode *clustering*, yaitu menganalisa data dengan mengelompokkan objek-objek kedalam sejumlah kelompok berdasarkan suatu kesamaan tertentu. Sedangkan algoritma pengelompokan data yang digunakan algoritma K-means berdasarkan pada jarak terdekat antar objeknya (*minimum distance*). Nilai centroid (titik pusat *cluster*) masing-masing hasil *clustering* selanjutnya akan digunakan untuk mengidentifikasi objek buah. K-means algorithm sangat

terkenal karena kemudahan dan kemampuannya untuk mengklaster data besar dengan sangat cepat. Oleh karena itu penulis memilih algoritma ini, agar pengenalan objek dapat dilakukan dengan cepat dan tepat.

1.2 Rumusan Masalah

Rumusan masalah yang dibahas dalam tugas akhir ini adalah bagaimana mengidentifikasi objek buah berdasarkan warna dengan *image clustering*.

1.3 Batasan Masalah

Ruang lingkup batasan dalam tugas akhir ini adalah:

1. Proses pembelajaran (*learning process*) dan proses identifikasi (*identification process*) menggunakan algoritma K-means dengan inisialisasi jumlah cluster adalah 3.
2. Objek yang digunakan adalah buah. Buah yang digunakan sebagai objek adalah buah dengan warna-warna dominan, seperti apel Washington, jeruk mandarin, mangga gadung, Pear, dan Anggur.
3. Objek yang digunakan sudah tersimpan dalam format JPG, dengan ukuran 32x32
4. Ciri yang digunakan yakni berupa ciri warna pada citra RGB (*Red, Green, dan Blue*).

1.4 Tujuan

Tujuan yang ingin dicapai dalam tugas akhir ini adalah:

1. Merancang bangun suatu aplikasi untuk mengidentifikasi buah berdasarkan ciri warnanya dengan *image clustering*.
2. Mempelajari dan menerapkan metode *Clustering* untuk pengenalan objek.
3. Mengetahui performa *image clustering* untuk pengenalan objek

1.5 Sistematika Penulisan

Sistematika penulisan laporan ini disusun agar dalam pembuatan laporan dapat lebih terstruktur dengan tahapan sistematika penulisan sebagai berikut:

BAB I PENDAHULUAN

Bab ini menjelaskan tentang latar belakang, rumusan masalah, batasan masalah, tujuan, dan sistematika penulisan.

BAB II LANDASAN TEORI

Bab ini membahas tentang pengolahan citra, *clustering*, karakteristik *clustering*, dan algoritma *clustering*.

BAB III METODOLOGI PENELITIAN

Bab ini menjelaskan tentang metodologi yang digunakan untuk menyelesaikan tugas akhir.

BAB IV ANALISA DAN PERANCANGAN

Bab ini membahas tentang analisis dan rancangan aplikasi pengenalan objek dengan menerapkan *imageclustering*.

BAB V IMPLEMENTASI DAN PENGUJIAN

Bab ini membahas tentang batasan implementasi, lingkungan implementasi, hasil implementasi dan hasil pengujian perangkat lunak, serta analisa terhadap hasil pengujian.

BAB VI PENUTUP

Bab ini berisi kesimpulan penelitian beserta saran-saran yang mungkin dapat dijadikan bahan pertimbangan yang bermanfaat bagi pengembangan sistem selanjutnya.

BAB II

LANDASAN TEORI

1.1 Pengolahan Citra

Pengolahan citra digital memiliki kegunaan dan spektrum aplikasi yang sangat luas. Teknologi pengolahan citra dapat dipergunakan di berbagai bidang seperti kedokteran, industri, pertanian, geologi, dan kelautan. Kehadiran teknologi pengolahan citra memberikan kemajuan yang luar biasa pada bidang-bidang tersebut. (Putra, 2010)

Pengolahan citra digital mampu mengidentifikasi jenis atau banyaknya objek pada suatu citra. Contoh aplikasinya adalah menghitung jumlah sel darah merah (eritrosit) yang rusak atau mengetahui kondisi sel darah, menghitung volume dari sampel citra gelembung sabun, dan menentukan penyebaran partikel pigmen pada citra kulit.

Pada contoh pemanfaatan pengolahan citra untuk menghitung atau mengetahui sel darah merah yang rusak dipergunakan teknik segmentasi citra dan *connected component labelling*, sehingga kondisi sel darah merah dapat diketahui.

1.1.1 Citra Digital

Secara umum, pengolahan citra digital merujuk pada pemrosesan gambar 2 (dua) dimensi dengan menggunakan komputer. Citra digital merupakan sebuah larik (*array*) yang berisi nilai-nilai real maupun kompleks yang direpresentasikan dengan deretan bit tertentu. (Putra, 2010)

Suatu citra dapat didefinisikan sebagai fungsi $f(x,y)$ berukuran M baris dan N kolom, dengan x dan y adalah koordinat spasial, dan amplitudo f di titik koordinat (x,y) dinamakan intensitas atau tingkat keabuan dari citra pada titik tersebut. Apabila nilai x , y , dan nilai amplitudo f secara keseluruhan berhingga (*finite*) dan bernilai diskrit maka dapat dikatakan bahwa citra tersebut adalah citra digital.

Citra digital yang berukuran $N \times M$ lazim dinyatakan dengan matriks yang berukuran N baris dan M kolom sebagai berikut:

$$f(x,y) \approx \begin{bmatrix} f(0,0) & f(0,1) & \dots & f(0,M) \\ f(1,0) & f(1,1) & \dots & f(1,M) \\ \vdots & \vdots & \vdots & \vdots \\ f(N-1,0) & f(N-1,1) & \dots & f(N-1,M-1) \end{bmatrix}$$

Indeks baris (i) dan indeks kolom (j) menyatakan suatu koordinat titik pada citra, sedangkan $f(i, j)$ merupakan intensitas (derajat keabuan) pada titik (i, j).

Masing-masing elemen pada citra digital (berarti elemen matriks) disebut *image element*, *picture element* atau pixel atau pel. Jadi, citra yang berukuran $N \times M$ mempunyai $N \times M$ buah pixel. Sebagai contoh, misalkan sebuah berukuran 256×256 pixel dan direpresentasikan secara numerik dengan matriks yang terdiri dari 256 buah baris (di-indeks dari 0 sampai 255) dan 256 buah kolom (di-indeks dari 0 sampai 255) seperti contoh berikut:

$$\begin{bmatrix} 0 & 134 & 145 & \dots & \dots & 231 \\ 0 & 167 & 201 & \dots & \dots & 197 \\ 220 & 187 & 189 & \dots & \dots & 120 \\ \vdots & \vdots & \vdots & \vdots & \vdots & \vdots \\ 221 & 219 & 210 & \dots & \dots & 156 \end{bmatrix}$$

Pixel pertama pada koordinat (0, 0) mempunyai nilai intensitas 0 yang berarti warna pixel tersebut hitam, pixel kedua pada koordinat (0, 1) mempunyai intensitas 134 yang berarti warnanya antara hitam dan putih, dan seterusnya.

1.1.2 Jenis Citra

Nilai suatu pixel memiliki nilai dalam rentang tertentu, dari nilai minimum sampai nilai maksimum. Jangkauan yang digunakan berbeda-beda tergantung dari jenis warnanya. Namun secara umum jangkauannya adalah 0 – 255. Citra dengan penggambaran seperti ini digolongkan ke dalam citra *integer*.

Berikut adalah jenis-jenis citra berdasarkan nilai pixelnya (Putra, 2010).

1. Citra Biner

Citra biner adalah citra digital yang hanya memiliki dua kemungkinan nilai pixel yaitu hitam dan putih. Citra biner juga disebut sebagai B&W (*Black and White*) atau monokrom. Hanya dibutuhkan 1 bit untuk mewakili nilai setiap pixel dari citra biner.

2. Citra *Grayscale*

Citra *grayscale* merupakan citra digital yang hanya memiliki satu nilai kanal pada setiap pixelnya. Warna yang dimiliki adalah warna dari hitam, keabuan dan putih. Tingkat keabuan disini merupakan warna abu dengan berbagai tingkatan dari hitam hingga mendekati putih. Citra *grayscale* memiliki kedalaman warna 8 bit (256 kombinasi warna keabuan).

3. Citra Warna (8 bit)

Setiap pixel pada citra warna (8-bit) hanya diwakili oleh 8 bit dengan jumlah warna maksimum yang dapat digunakan adalah 256 warna. Ada dua jenis citra warna 8 bit. Pertama, citra warna 8 bit dengan menggunakan palet warna 256 dengan setiap paletnya memiliki pemetaan nilai (*colormap*) RGB tertentu. Kedua, setiap pixel memiliki format 8 bit *truecolor*

4. Citra Warna (16 bit)

Citra warna 16 bit (disebut juga citra *highcolor*) dengan setiap pixelnya diwakili dengan 2 byte memori (16 bit). Warna 16 bit memiliki 65.536 warna.

5. Citra Warna (24 bit)

Setiap pixel dari citra warna 24 bit diwakili oleh 24 bit sehingga total 16.777.216 variasi warna. Variasi ini lebih dari cukup untuk memvisualisasikan seluruh warna yang dapat dilihat penglihatan manusia. Penglihatan manusia dipercaya hanya dapat membedakan hingga 10 juta warna saja.

Setiap poin informasi pixel (RGB) disimpan ke dalam 1 byte data. 8 bit pertama menyimpan nilai biru, diikuti dengan nilai hijau pada 8 bit kedua dan pada 8 bit terakhir merupakan warna merah.

1.1.3 Format File Citra

Format file citra standar yang digunakan saat ini terdiri dari beberapa jenis. Format-format ini digunakan dalam menyimpan citra dalam sebuah *file*. Setiap format memiliki karakteristik masing-masing.

Berikut ini adalah penjelasan beberapa format umum digunakan saat ini (Putra, 2010).

1. Bitmap (.bmp)

Format ini adalah format penyimpanan standar tanpa kompresi yang umum dapat digunakan untuk menyimpan citra biner hingga citra warna.

2. *Portable Network Graphics* (.png)

Format ini adalah format penyimpanan citra terkompresi. Format ini dapat digunakan pada citra *grayscale*, citra dengan palet warna, dan juga citra *fullcolor*. Format .png juga mampu menyimpan informasi hingga kanal alpha dengan penyimpanan sebesar 1 hingga 16 bit per kanal.

3. JPEG (.jpg)

Format ini adalah format yang sangat umum digunakan saat ini khususnya untuk transmisi citra. Format ini digunakan untuk menyimpan citra hasil kompresi dengan metode JPEG.

4. GIF (.gif)

Format ini dapat digunakan pada citra warna dengan palet 8 bit. Penggunaan umumnya pada aplikasi *web*. Kualitas yang rendah menyebabkan format ini tidak terlalu populer di kalangan peneliti pengolahan citra digital.

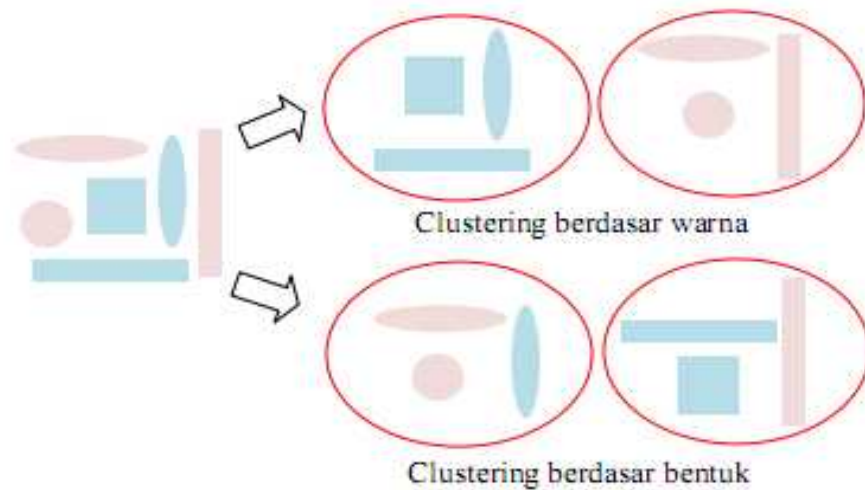
1.2 *Clustering*

Clustering berarti penyatuan sekelompok data yang mempunyai korelasi atau karakteristik sejenis atau sekelompok data yang memiliki kemiripan. *Clustering* juga merupakan suatu cara menganalisa data dengan cara mengelompokkan objek kedalam kelompok-kelompok berdasar suatu kesamaan tertentu (Ming dkk, 2004). *Clustering* berbeda dengan proses group data. Dalam group, pengelompokan data harus sama jadi sudah dapat dipastikan yang termasuk dalam group itu dan memiliki karakter yang sama. *Clustering* banyak digunakan dalam kasus pemetaan (*mapping*) seperti yang didefinisikan $f:D \rightarrow C$, dari beberapa data $D=\{t_1, t_2, t_3, \dots, t_n\}$ kedalam beberapa *cluster* $C=\{c_1, c_2, c_3, \dots, c_n\}$ berdasarkan kesamaan antar data t_1 (Barakbah dkk, 2004).

Segmentasi berbasis *cluster* menggunakan data multidimensi untuk mengelompokkan pixel citra ke dalam beberapa *cluster*. Pada umumnya pixel di-*cluster* berdasarkan kedekatan jarak antar pixel. Segmentasi berbasis *cluster* ini mulai populer sejak diimplementasikan pada aplikasi OCR (*Optical Character Recognition*), pengenalan sidik jari, hingga *remote sensing*. Keberhasilan dari proses segmentasi berbasis *cluster* ditentukan dari keberhasilan dalam mengelompokkan fitur-fitur yang berdekatan ke dalam satu *cluster*.

Metode-metode dalam segmentasi berbasis *cluster* di antaranya adalah iterasi, K-Means, fuzzy C-Means, dan jaringan syaraf Kohonen.

Berikut ilustrasi *clustering*:



Gambar 2.1 Ilustrasi Clustering

1.2.1 Karakteristik *Clustering*

Karakteristik *clustering* dibagi menjadi 4, yaitu:

1. *Partitioning clustering*

Partitioning clustering disebut juga *exclusive clustering*, dimana setiap data harus termasuk ke *cluster* tertentu. Karakteristik tipe ini juga memungkinkan bagi setiap data yang termasuk *cluster* tertentu pada suatu tahapan proses, pada tahap berikutnya berpindah ke *cluster* yang lain. Contoh : K-Means, *residual analysis*.

2. *Hierarchical clustering*

Pada *hierarchical clustering*, setiap data harus termasuk ke *cluster* tertentu. Suatu data yang termasuk ke *cluster* tertentu pada suatu tahapan proses, tidak dapat berpindah ke *cluster* lain pada tahapan berikutnya. Contoh: Single Linkage, Centroid Linkage, Complete Linkage, Average Linkage.

3. *Overlapping clustering*

Pada *overlapping clustering*, setiap data memungkinkan termasuk ke beberapa *cluster*. Data mempunyai nilai keanggotaan (*membership*) pada beberapa *cluster*. Contoh: Fuzzy C-means, Gaussian Mixture.

4. *Hybrid*

Karakteristik *hybrid* adalah mengawinkan karakteristik dari *partitioning*, *overlapping* dan *hierarchical*.

1.2.2 Algoritma *Clustering*

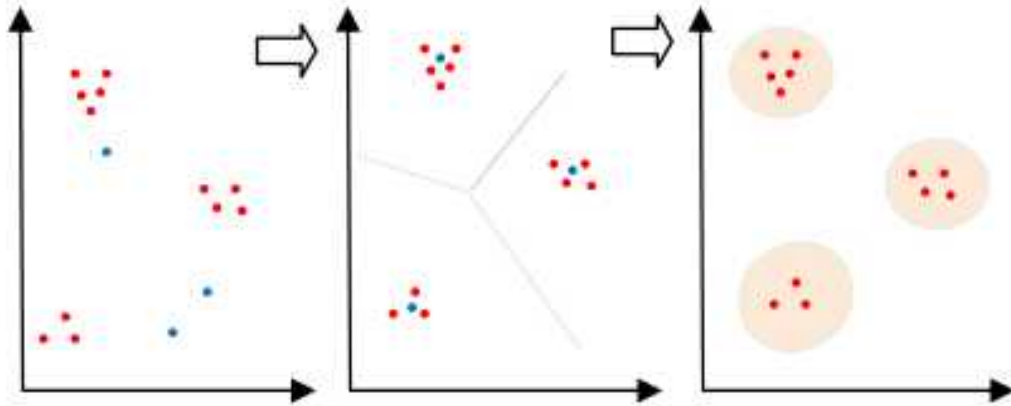
Ada beberapa algoritma yang sering digunakan dalam *clustering*, yaitu:

1. *K-Means*

Algoritma *K-Means* adalah metode *clustering* berbasis jarak yang membagi data ke dalam sejumlah *cluster* dan algoritma ini hanya bekerja pada atribut numerik. Pada dasarnya penggunaan algoritma dalam melakukan proses *clustering* tergantung dari data yang ada dan konklusi yang ingin dicapai (Barakbah, 2006) termasuk *partitioning clustering* yang memisahkan data ke k daerah bagian yang terpisah. *K-meansalgorithm* sangat terkenal karena kemudahan dan kemampuannya untuk mengklaster data besar dengan sangat cepat. Sesuai dengan karakteristik *partitioning clustering*, Setiap data harus termasuk ke *cluster* tertentu, dan memungkinkan bagi setiap data yang termasuk *cluster* tertentu pada suatu tahapan proses, pada tahapan berikutnya berpindah ke *cluster* yang lain.

Algoritma *K-Means* :

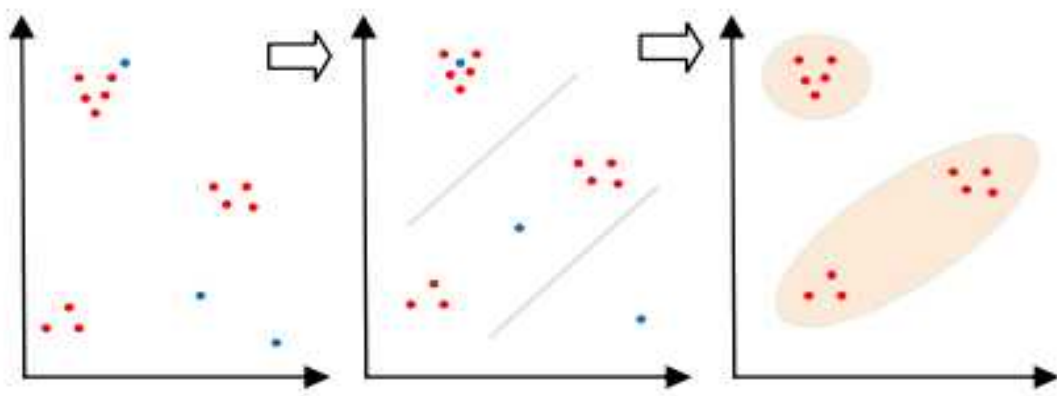
- a. Menentukan k sebagai jumlah *cluster* yang ingin dibentuk.
- b. Membangkitkan *centroids* (titik pusat *cluster*) awal secara random.
- c. Menghitung jarak setiap data ke masing-masing *centroid*.
- d. Setiap data memilih *centroid* yang terdekat.
- e. Menentukan posisi *centroid* baru dengan cara menghitung nilai rata-rata dari data-data yang memilih pada *centroid* yang sama.
- f. Kembali ke langkah 3 jika posisi *centroids* baru dengan *centroids* lama tidak sama.



Gambar 2.2 Ilustrasi Algoritma *K-Means*

Karakteristik *K-Means* :

- K-means* sangat cepat dalam proses *clustering*
- K-means* sangat sensitif pada pembangkitan *centroids* awal secara random
- Memungkinkan suatu *cluster* tidak mempunyai anggota
- Hasil *clustering* dengan *K-means* bersifat tidak unik (selalu berubah-ubah terkadang baik, terkadang jelek).
- K-means* sangat sulit untuk mencapai global optimum



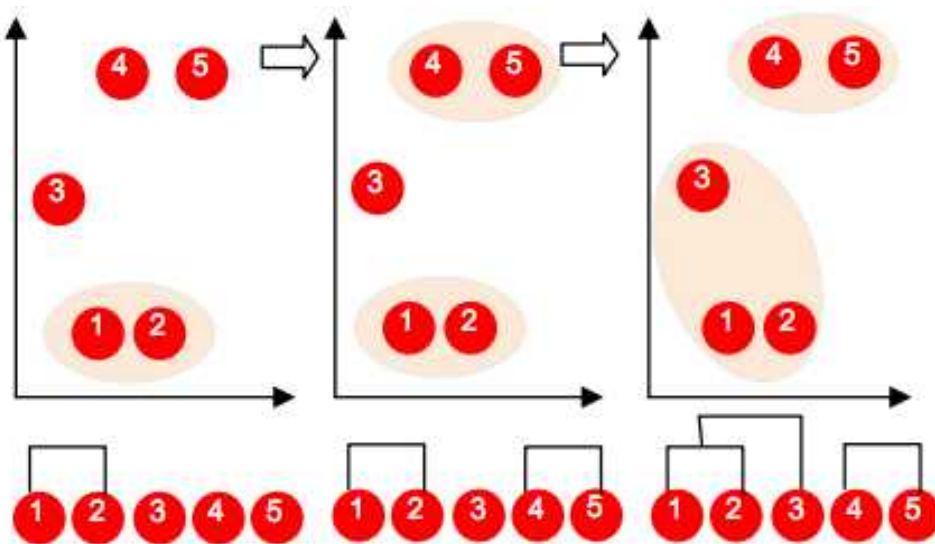
Gambar 2.3 Ilustrasi Kelemahan *K-Means*

2. *Hierarchical Clustering*

Dengan metode ini, data tidak langsung dikelompokkan kedalam beberapa *cluster* dalam 1 tahap, tetapi dimulai dari 1 *cluster* yang mempunyai kesamaan, dan berjalan seterusnya selama beberapa iterasi, hingga terbentuk beberapa *cluster* tertentu (Barakbah dkk, 2004).

Algoritma *Hierarchical clustering*:

- Menentukan k sebagai jumlah *cluster* yang ingin dibentuk
- Setiap data dianggap sebagai *cluster*. Kalau N = jumlah data dan c = jumlah *cluster*, berarti ada $c=N$.
- Menghitung jarak antar *cluster*
- Cari 2 *cluster* yang mempunyai jarak antar *cluster* yang paling minimal dan gabungkan (berarti $c=c-1$).
- Jika $c > k$, kembali ke langkah 3



Gambar 2.4 Ilustrasi Algoritma *Hierarchical Clustering*

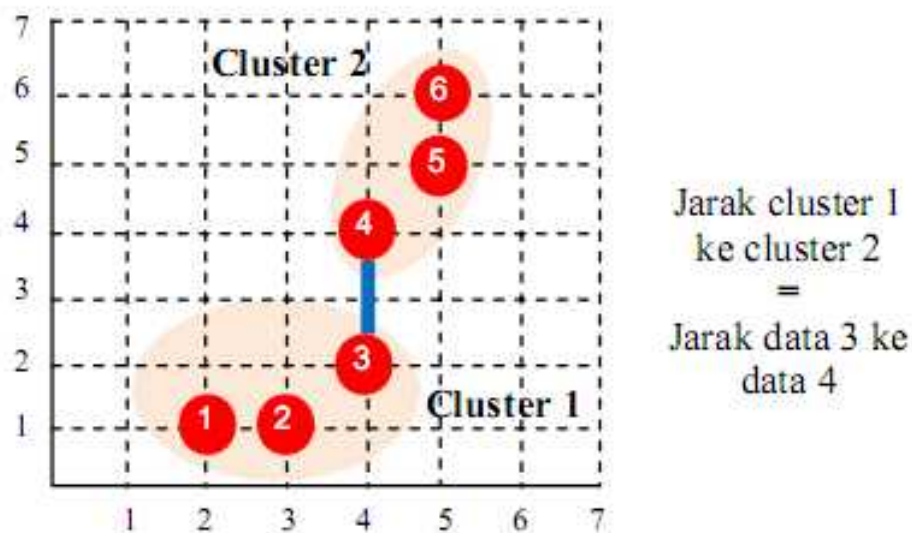
Algoritma *hierarchical clustering* banyak diaplikasikan pada metode peng-cluster-an berikut:

1. *Single Linkage Hierarchical Method (SLHM)*

Single Linkage adalah proses peng-*cluster*-an yang didasarkan pada jarak terdekat antar obyeknya (*minimum distance*). Metode SLHM sangat bagus untuk melakukan analisis pada tiap tahap pembentukan *cluster*. Metode ini juga sangat cocok untuk dipakai pada kasus *shape independent clustering*, karena kemampuannya untuk membentuk pola (*pattern*) tertentu dari *cluster*.

Algoritma *Single Linkage Hierarchical Method*:

- Diasumsikan setiap data dianggap sebagai *cluster*. Kalau n =jumlah data dan c =jumlah *cluster*, berarti ada $c=n$.
- Menghitung jarak antar *cluster* dengan *Euclidean distance*.
- Mencari 2 *cluster* yang mempunyai jarak antar *cluster* yang paling minimal dan digabungkan (*merge*) kedalam *cluster* baru (sehingga $c=c-1$)
- Kembali ke langkah 3, dan diulangi sampai dicapai *cluster* yang diinginkan.



Gambar 2.5 Ilustrasi *Single Linkage*

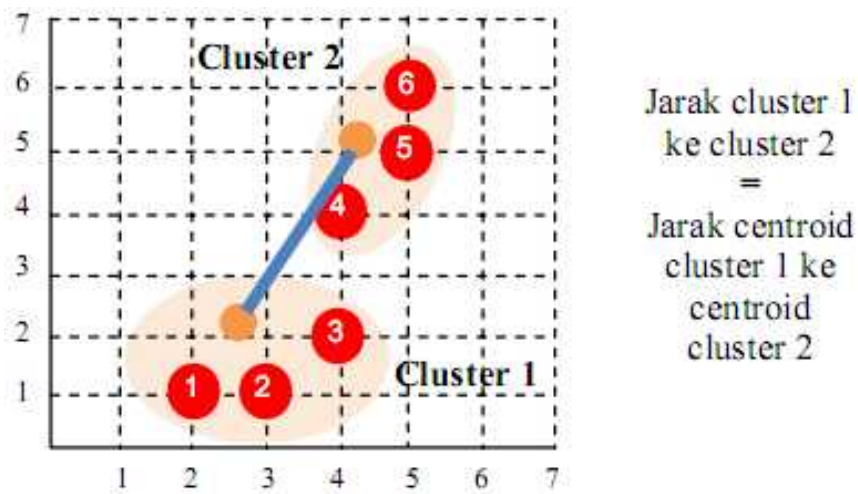
2. *Centroid Linkage Hierarchical Method*

Centroid Linkage adalah proses peng-*cluster*-an yang didasarkan pada jarak antar *centroid*-nya (Barakbah dkk, 2004). Metode ini bagus untuk memperkecil *variance within*

cluster karena melibatkan *centroid* pada saat penggabungan antar *cluster*. Metode ini juga baik untuk data yang mengandung outlier.

Algoritma *Centroid Linkage Hierarchical Method*:

- Diasumsikan setiap data dianggap sebagai *cluster*. Kalau n = jumlah data dan c = jumlah *cluster*, berarti ada $c = n$.
- Menghitung jarak antar *cluster* dengan *Euclidean distance*.
- Mencari 2 *cluster* yang mempunyai jarak centroid antar *cluster* yang paling minimal dan digabungkan (*merge*) kedalam *cluster* baru (sehingga $c = c - 1$).
- Kembali ke langkah 3, dan diulangi sampai dicapai *cluster* yang diinginkan.



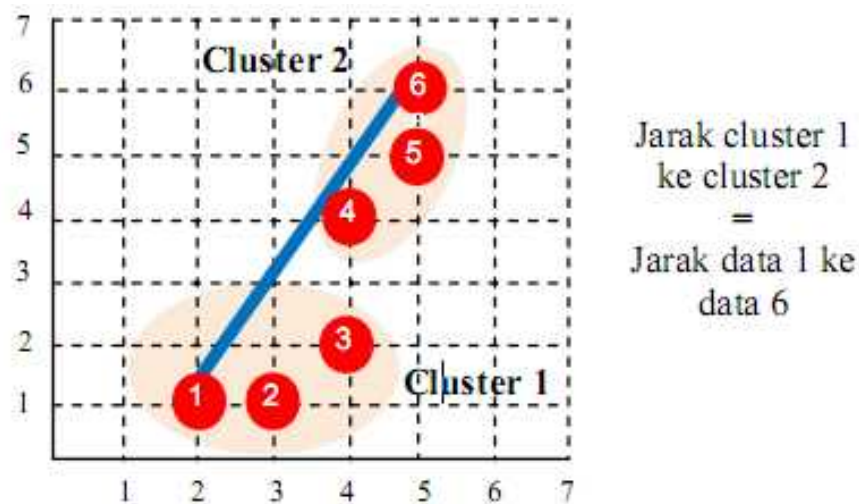
Gambar 2.6 Ilustrasi *Centroid Linkage*

3. *Complete Linkage Hierarchical Method*

Complete Linkage adalah proses peng-cluster-an yang didasarkan pada jarak terjauh antar obyeknya (*maximum distance*) (Barakbah dkk, 2004). Metode ini baik untuk kasus *clustering* dengan *normal data set distribution*. Akan tetapi, metode ini tidak cocok untuk data yang mengandung outlier.

Algoritma *Complete Linkage Hierarchical Method* :

- Diasumsikan setiap data dianggap sebagai *cluster*. Kalau n =jumlah data dan c =jumlah *cluster*, berarti ada $c=n$.
- Menghitung jarak antar *cluster* dengan *Euclidean distance*.
- Mencari 2 *cluster* yang mempunyai jarak antar *cluster* yang paling maksimal/terjauh dan digabungkan (*merge*) kedalam *cluster* baru (sehingga $c=c-1$)
- Kembali ke langkah 3, dan diulangi sampai dicapai *cluster* yang diinginkan.



Gambar 2.7 Ilustrasi *Complete Linkage*

4. *Average Linkage Hierarchical Method*

Average Linkage adalah proses peng-*cluster*-an yang didasarkan pada jarak rata-rata antar obyeknya (*average distance*) (Barakbah dkk, 2004). Metode ini relatif yang terbaik dari metode-metode *hierarchical*. Namun, ini harus dibayar dengan waktu komputasi yang paling tinggi dibandingkan dengan metode-metode *hierarchical* yang lain.

1.2.3 *Euclidean Distance*

Kedekatan atau kemiripan (*similarity*) suatu objek dengan objek lain atau dengan pusat *cluster* dihitung dengan menggunakan fungsi jarak. Pada umumnya fungsi jarak yang digunakan adalah jarak Euclidean (*Euclidean distance*) untuk menghitung kemiripan tersebut (Barakbah dkk, 2004) (Putra, 2010). *Euclidean distance* merupakan jarak antara dua titik sebagai mana

diukur dengan menggunakan penggaris, didasarkan pada rumus Pitagoras. Dalam koordinat Kartesius, jika $p = (p_1, p_2, \dots, p_n)$ dan $q = (q_1, q_2, \dots, q_n)$ merupakan dua buah titik dalam Euclidean n -space, maka jarak antara p dan q dapat dihitung dengan rumus:

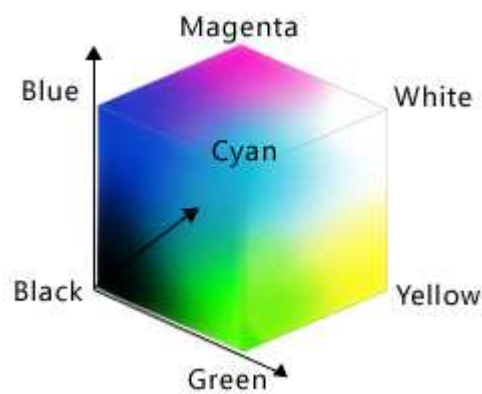
$$d(p, q) = d(q, p) = \sqrt{(q_1 - p_1)^2 + (q_2 - p_2)^2 + \dots + (q_n - p_n)^2} = \sqrt{\sum_{i=1}^n (q_i - p_i)^2} \quad (2.1)$$

1.3 Konsep Warna

Warna pada dasarnya merupakan hasil persepsi dari cahaya dalam spektrum wilayah yang terlihat oleh retina mata, dan memiliki panjang gelombang antara 400 nm sampai dengan 700 nm (Poynton, 1997 dikutip dari Lestari, 2003). Ruang warna atau yang sering disebut sebagai model warna merupakan sebuah cara atau metode untuk mengatur, membuat dan memvisualisasikan warna. Salah satu jenis ruang warna yaitu ruang warna citra RGB.

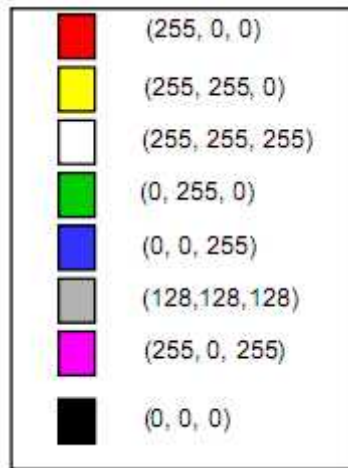
1.3.1 RGB (*Red Green Blue*)









Citra berwarna yang selama ini biasa kita kenal umumnya memiliki ruang warna RGB. Ruang warna RGB dapat divisualisasikan sebagai sebuah kubus seperti pada gambar 2.8, dengan tiga sumbunya mewakili komponen warna merah (*red*) R, hijau (*green*) G, biru (*blue*) B. Salah satu pojok alas kubus ini menyatakan warna hitam ketika $R = G = B = 0$, sedangkan pojok atasnya yang berlawanan menyatakan warna putih ketika $R = G = B = 255$ (untuk sistem warna 8 bit bagi setiap komponennya). RGB sering digunakan di dalam sebagian besar aplikasi komputer karena dengan ruang warna ini, tidak diperlukan transformasi untuk menampilkan informasi di layar monitor. Alasan di atas juga menyebabkan RGB banyak dimanfaatkan sebagai ruang warna dasar bagi sebagian besar aplikasi.



Gambar 2.8 Ruang warna RGB (Sumber : *MATLAB Image Processing Toolbox, The MathWorks Inc.* dikutip dari Lestari, 2003)

Jika warna-warna pokok (RGB) tersebut digabungkan, maka akan menghasilkan warna lain. Tiap-tiap warna memiliki nilai 256 (8 bit).



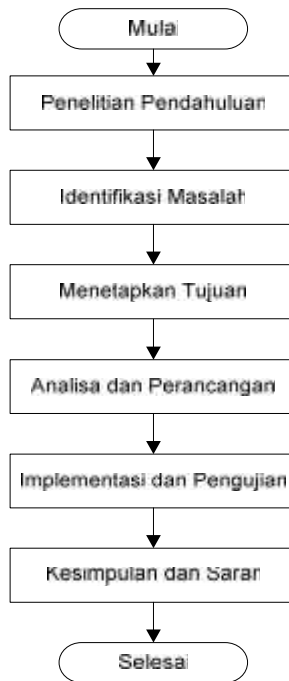
	(255, 0, 0)
	(255, 255, 0)
	(255, 255, 255)
	(0, 255, 0)
	(0, 0, 255)
	(128, 128, 128)
	(255, 0, 255)
	(0, 0, 0)

Gambar 2.9 Konsep Warna

BAB III

METODOLOGI PENELITIAN

Pada bab ini diuraikan langkah-langkah yang digunakan selama penelitian berlangsung. Gambaran langkah-langkah tersebut secara garis besar dapat dilihat pada *flowchart* berikut:



Gambar 3.1 *Flowchart* langkah-langkah penelitian

Langkah-langkah yang dilakukan dalam penelitian ini meliputi penelitian pendahuluan, mengidentifikasi masalah, menetapkan tujuan, mengumpulkan data, menganalisa data yang telah dikumpulkan dan kebutuhan dari aplikasi yang akan dirancang bangun, perancangan, penulisan kode program dan pengujian. Apabila suatu tahap yang dilakukan memiliki kekurangan maka harus kembali ke tahap sebelumnya untuk menyelesaikannya, apabila telah selesai maka dapat melanjutkan ke tahap selanjutnya.

1.1 Penelitian Pendahuluan

Penelitian pendahuluan dilakukan untuk mencari informasi mengenai penelitian yang berhubungan dengan pengenalan objek dan metode *image clustering*. Pengumpulan informasi dilakukan melalui studi pustaka, dengan cara mempelajari konsep *clustering* dan *image training* yang diperoleh dari buku-buku, artikel-artikel, jurnal-jurnal dan media lainnya.

1.2 Identifikasi Masalah

Dengan memanfaatkan informasi yang diperoleh dari studi pustaka yang telah dilakukan, selanjutnya dilakukan identifikasi masalah. Pada tugas akhir ini masalah yang diidentifikasi adalah bagaimana merancang bangun suatu aplikasi yang dapat mengidentifikasi objek buah berdasarkan warna dengan metode *image clustering*.

1.3 Penetapan Tujuan

Penetapan tujuan sangat diperlukan untuk menjawab permasalahan yang berhasil diidentifikasi. Tujuan ditetapkan dengan cara mengetahui dan menentukan apa saja yang perlu dipertahankan, ditingkatkan, dihilangkan, dievaluasi dan diperbarui agar masalah yang ada dapat teratasi.

1.4 Pengumpulan Data

Pengumpulan data bertujuan untuk mendapatkan data yang dibutuhkan untuk membangun suatu aplikasi yang dapat mengidentifikasi objek buah berdasarkan warna dengan metode *image clustering*.

1.5 Analisa dan Perancangan

Analisa perangkat lunak dilakukan untuk mengidentifikasi kebutuhan dan fungsi perangkat lunak serta memodelkan perangkat lunak yang akan dibangun. Alat bantu yang digunakan adalah *flowchart*.

Perancangan perangkat lunak yang dilakukan meliputi perancangan basis data dan antar muka, termasuk struktur menu dan tampilan, terkait proses pembelajaran (*learning process*) dan proses identifikasi (*identification process*).

1.6 Implementasi dan Pengujian

Setelah analisa dan perancangan perangkat lunak, dilakukan penulisan kode program. Selanjutnya dilakukan pengujian terhadap hasil implementasi meliputi pengujian terhadap proses pembelajaran (*learning process*) dan proses identifikasi (*identification process*), untuk mengetahui apakah aplikasi dapat mengidentifikasi dan mengenali jenis buah apa saja yang terdapat pada citra.

1.7 Kesimpulan dan Saran

Kesimpulan diambil setelah seluruh langkah-langkah penelitian selesai dilakukan. Kesimpulan yang diperoleh dari penelitian ini meliputi hasil yang diperoleh termasuk performa dari aplikasi yang telah dirancang bangun. Saran yang dapat diberikan merupakan harapan di masa datang untuk pengembangan aplikasi selanjutnya.

BAB IV

ANALISA DAN PERANCANGAN

1.1 Analisa

Pada sub bab ini diuraikan deskripsi aplikasi yang dirancang bangun beserta pemodelannya.

1.1.1 Deskripsi Umum

Aplikasi yang dirancang bangun dalam Tugas Akhir ini adalah aplikasi untuk mengidentifikasi objek buah berdasarkan warna dengan *image clustering*. Ciri dari objek buah yang digunakan yakni berupa ciri warna pada citra RGB (*Red, Green, dan Blue*).

Dalam aplikasi ini, terdapat dua buah proses utama yaitu:

1. Proses pembelajaran (*learning process*) menggunakan algoritma K-means dengan inisialisasi awal jumlah *cluster* adalah 3. Jumlah *cluster* didasarkan pada citra buah tunggal yang digunakan untuk pembelajaran. Citra buah tunggal terdiri atas 3 bagian, yaitu buah, bayangan dari buah, dan latar belakang (*background*). Dalam proses pembelajaran, diperoleh nilai centroid (titik pusat *cluster*) dari bagian buah untuk diberi label sesuai dengan nama buah yang bersangkutan dan disimpan ke dalam basis data.
2. Proses identifikasi (*identification process*) juga menggunakan algoritma K-means. Citra yang digunakan untuk identifikasi merupakan citra sekumpulan buah. Dalam proses identifikasi, dilakukan *imageclustering* terhadap citra tersebut dan didapatkan nilai centroid dari masing-masing *cluster*. Nilai centroid ini dihitung kedekatannya dengan Euclidean *distance* dengan nilai centroid dari proses pembelajaran yang sebelumnya telah disimpan ke dalam basis data. Hasil proses ini adalah jumlah jenis buah yang ada pada gambar beserta namanya.

1.1.2 Citra Masukan

Berdasarkan proses yang menggunakannya, citra masukan dibedakan menjadi 2 yaitu:

1. Gambar untuk pembelajaran, berupa citra buah tunggal dengan latar belakang (*background*) berwarna putih.
2. Gambar untuk identifikasi, berupa citra sekumpulan buah dengan latar belakang (*background*) berwarna putih.

1.1.3 Proses Pembelajaran

Dalam proses pembelajaran, dilakukan sejumlah langkah sebagai berikut:

1. Ekstraksi warna untuk memperoleh nilai RGB dari pixel-pixel yang ada pada gambar.

Misalkan terdapat gambar buah apel, sampel yang diambil ada lima gambar apel dengan bentuk dan ukuran yang berbeda. Setelah diekstraksi gambar-gambar apel tersebut, maka akan menghasilkan nilai RGB masing-masing pixel pada gambar apel tersebut. Misalkan gambar apel berukuran pixel 4 x 3,

1	2	3
4	5	6
7	8	9
10	11	12

Gambar 4.1 dengan pixel 4 x 3

yang akan menghasilkan RGB masing-masing pixel sebagai berikut :

Tabel 4.1Tabel RGB Pixel

Pixel	R	G	B
1	44	39	43
2	135	124	128
3	183	174	177
4	188	182	184
5	191	189	190
6	195	193	196
7	196	194	197
8	196	194	197
9	194	193	198
10	191	192	196
11	160	156	170
12	160	155	162

2. Kelompokkan nilai RGB menggunakan algoritma K-means. Pada langkah ini dilakukan hal-hal berikut:

- Inisialisasi jumlah *cluster* yang ingin dibentuk sebanyak 3, yang merepresentasikan buah, bayangan dari buah dan latar belakang (*background*). Tujuan dari penentuan jumlah *cluster* adalah untuk menentukan jumlah inisial pusat *cluster* yang akan dibangkitkan.
- Random nilai RGB dari 3 *centroid* (titik pusat *cluster*) awal berdasarkan jangkauan (*range*) nilai minimum dan maksimum RGB dari pixel-pixel yang ada pada gambar.

$$R = 44 \text{ s/d } 196$$

$$G = 39 \text{ s/d } 194$$

$$B = 43 \text{ s/d } 198$$

Misalkan diperoleh nilai *centroid* / pusat *cluster* dari masing-masing *cluster* seperti pada tabel 4.2

Tabel 4.2 Tabel RGB *Centroid* / pusat *Cluster*

<i>Cluster</i>	R	G	B
1	44	39	43
2	196	194	197
3	160	156	170

- Hitung jarak nilai RGB setiap pixel terhadap masing-masing *centroid* / pusat *cluster* tersebut dengan menggunakan rumus 2.1 yaitu Rumus *Euclidean distance*. Jarak RGB pixel pertama dengan pusat *cluster* adalah :

$$c(1,1) = \sqrt{(44 - 44)^2 + (39 - 39)^2 + (43 - 43)^2} = 0$$

$$c(1,2) = \sqrt{(44 - 196)^2 + (39 - 194)^2 + (43 - 197)^2} = 266,17$$

$$c(1,3) = \sqrt{(44 - 160)^2 + (39 - 156)^2 + (43 - 170)^2} = 208,02$$

Jarak RGB pixel ke-2 dengan pusat *cluster* adalah :

$$c(2,1) = \sqrt{(135 - 44)^2 + (124 - 39)^2 + (128 - 43)^2} = 150,77$$

$$c(2,2) = \sqrt{(135 - 196)^2 + (124 - 194)^2 + (128 - 197)^2} = 115,68$$

$$c(2,3) = \sqrt{(135 - 160)^2 + (124 - 156)^2 + (128 - 170)^2} = 58,42$$

Jarak RGB pixel ke-3 dengan pusat *cluster* adalah :

$$c(3,1) = \sqrt{(183 - 44)^2 + (174 - 39)^2 + (177 - 43)^2} = 235,5$$

$$c(3,2) = \sqrt{(183 - 196)^2 + (174 - 194)^2 + (177 - 197)^2} = 26,70$$

$$c(3,3) = \sqrt{(183 - 160)^2 + (174 - 156)^2 + (177 - 170)^2} = 30,03$$

Jarak RGB pixel ke-4 dengan pusat *cluster* adalah :

$$c(4,1) = \sqrt{(188 - 44)^2 + (182 - 39)^2 + (184 - 43)^2} = 247,12$$

$$c(4,2) = \sqrt{(188 - 196)^2 + (182 - 194)^2 + (184 - 197)^2} = 19,42$$

$$c(4,3) = \sqrt{(188 - 160)^2 + (182 - 156)^2 + (184 - 170)^2} = 40,69$$

Jarak RGB pixel ke-5 dengan pusat *cluster* adalah :

$$c(5,1) = \sqrt{(191 - 44)^2 + (189 - 39)^2 + (190 - 43)^2} = 256,35$$

$$c(5,2) = \sqrt{(191 - 196)^2 + (189 - 194)^2 + (190 - 197)^2} = 9,95$$

$$c(5,3) = \sqrt{(191 - 160)^2 + (189 - 156)^2 + (190 - 170)^2} = 49,5$$

Jarak RGB pixel ke-6 dengan pusat *cluster* adalah :

$$c(6,1) = \sqrt{(195 - 44)^2 + (193 - 39)^2 + (196 - 43)^2} = 264,43$$

$$c(6,2) = \sqrt{(195 - 196)^2 + (193 - 194)^2 + (196 - 197)^2} = 1,73$$

$$c(6,3) = \sqrt{(195 - 160)^2 + (193 - 156)^2 + (196 - 170)^2} = 57,18$$

Jarak RGB pixel ke-7 dengan pusat *cluster* adalah :

$$c(7,1) = \sqrt{(196 - 44)^2 + (194 - 39)^2 + (197 - 43)^2} = 266,17$$

$$c(7,2) = \sqrt{(196 - 196)^2 + (194 - 194)^2 + (197 - 197)^2} = 0$$

$$c(7,3) = \sqrt{(196 - 160)^2 + (194 - 156)^2 + (197 - 170)^2} = 58,9$$

Jarak RGB pixel ke-8 dengan pusat *cluster* adalah :

$$c(8,1) = \sqrt{(196 - 44)^2 + (194 - 39)^2 + (197 - 43)^2} = 266,17$$

$$c(8,2) = \sqrt{(196 - 196)^2 + (194 - 194)^2 + (197 - 197)^2} = 0$$

$$c(8,3) = \sqrt{(196 - 160)^2 + (194 - 156)^2 + (197 - 170)^2} = 58,9$$

Jarak RGB pixel ke-9 dengan pusat *cluster* adalah :

$$c(9,1) = \sqrt{(194 - 44)^2 + (193 - 39)^2 + (198 - 43)^2} = 265,03$$

$$c(9,2) = \sqrt{(194 - 196)^2 + (193 - 194)^2 + (198 - 197)^2} = 2,45$$

$$c(9,3) = \sqrt{(194 - 160)^2 + (193 - 156)^2 + (198 - 170)^2} = 57,52$$

Jarak RGB pixel ke-10 dengan pusat *cluster* adalah :

$$c(10,1) = \sqrt{(191 - 44)^2 + (192 - 39)^2 + (196 - 43)^2} = 261,58$$

$$c(10,2) = \sqrt{(191 - 196)^2 + (192 - 194)^2 + (196 - 197)^2} = 5,48$$

$$c(10,3) = \sqrt{(191 - 160)^2 + (192 - 156)^2 + (196 - 170)^2} = 54,16$$

Jarak RGB pixel ke-11 dengan pusat *cluster* adalah :

$$c(11,1) = \sqrt{(160 - 44)^2 + (156 - 39)^2 + (170 - 43)^2} = 208,02$$

$$c(11,2) = \sqrt{(160 - 196)^2 + (156 - 194)^2 + (170 - 197)^2} = 58,9$$

$$c(11,3) = \sqrt{(160 - 160)^2 + (156 - 156)^2 + (170 - 170)^2} = 0$$

Jarak RGB pixel ke-12 dengan pusat *cluster* adalah :

$$c(12,1) = \sqrt{(160 - 44)^2 + (155 - 39)^2 + (162 - 43)^2} = 202,66$$

$$c(12,2) = \sqrt{(160 - 196)^2 + (155 - 194)^2 + (162 - 197)^2} = 63,58$$

$$c(12,3) = \sqrt{(160 - 160)^2 + (155 - 156)^2 + (162 - 170)^2} = 8,06$$

Berdasarkan hasil perhitungan di atas maka diperoleh data :

Tabel 4.3 Hasil Perhitungan Jarak Setiap data

Pixel	R	G	B	C1	C2	C3
1	44	39	43	0	266,17	208,02
2	135	124	128	150,77	115,68	58,42
3	183	174	177	235,5	26,70	30,03
4	188	182	184	247,12	19,42	40,69
5	191	189	190	256,35	9,95	49,5
6	195	193	196	264,43	1,73	57,18
7	196	194	197	266,17	0	58,9
8	196	194	197	266,17	0	58,9
9	194	193	198	265,03	2,45	57,52
10	191	192	196	261,58	5,48	54,16
11	160	156	170	208,02	58,9	0
12	160	155	162	202,66	63,58	8,06

- d. *Object clustering* dilakukan berdasarkan hasil perhitungan di atas, dimana setiap pixel menjadi anggota suatu *cluster* yang *Euclidean distance* dari nilai RGB pixel tersebut terhadap centroid-nya minimum.

Tabel 4.4 Tabel RGB Centroid Cluster

Pixel	R	G	B	C1	C2	C3
1	44	39	43			
2	135	124	128			
3	183	174	177			
4	188	182	184			
5	191	189	190			
6	195	193	196			
7	196	194	197			
8	196	194	197			
9	194	193	198			
10	191	192	196			
11	160	156	170			
12	160	155	162			

- e. Hitung posisi pusat *cluster* baru berdasarkan rata-rata anggota *cluster*.

Untuk *cluster* pertama, ada 1 data yaitu data pertama, sehingga

$$R = 44, G = 39, B = 43$$

Untuk *cluster* kedua, ada 8 data yaitu data ke-3, 4, 5, 6, 7, 8, 9, dan data ke-10, sehingga :

$$R = \frac{183 + 188 + 191 + 195 + 196 + 196 + 194 + 191}{8} = 191,75$$

$$G = \frac{174 + 182 + 189 + 193 + 194 + 194 + 193 + 192}{8} = 188,87$$

$$B = \frac{177 + 184 + 190 + 196 + 197 + 197 + 198 + 196}{8} = 191,87$$

Untuk *cluster* ketiga, ada 3 data yaitu data ke-2, 11, dan data ke-12, sehingga :

$$R = \frac{135 + 160 + 160}{3} = 151,67$$

$$G = \frac{124 + 156 + 155}{3} = 145$$

$$B = \frac{128 + 170 + 162}{3} = 153,33$$

- f. Periksa apakah pixel-pixel mengalami perpindahan *cluster*. Jika masih ada, maka lanjutkan ke langkah c. Jika tidak, berarti komputasi telah mencapai stabilitas.
3. Berdasarkan 3 centroid tersebut, eliminasi *cluster* bayangan dari buah, yaitu *cluster* dengan jumlah anggota paling sedikit. Hal ini dikarenakan *cluster* dengan jumlah anggota kecil merepresentasikan nilai *noise*.

Cluster bayangan yang dihapus adalah cluster 1 karena mempunyai 1 anggota.

4. Berdasarkan 2 sisa centroid yang ada, eliminasi latar belakang (*background*), yaitu *cluster* dengan jarak terdekat antara nilai RGB-nya terhadap nilai RGB latar belakang (*background*) berwarna putih (255,255,255) menggunakan Euclidean *distance*.

Cluster 2 :

$$d = \sqrt{(255 - 191,75)^2 + (255 - 188,87)^2 + (255 - 191,87)^2} = 111,17$$

Cluster 3 :

$$d = \sqrt{(255 - 151,67)^2 + (255 - 145)^2 + (255 - 153,33)^2} = 181,97$$

Berdasarkan perhitungan di atas, cluster yang di eliminasi adalah cluster 2, karena memiliki jarak terdekat dengan RGB latar belakang.

5. Simpan nilai centroid yang tersisa (centroid buah yang dicari) beserta label namanya ke dalam basis data. Nilai label dientri oleh pengguna.

Cluster yang tersisa adalah cluster 3 dengan nilai centroid R = 151,67 G = 145 dan B = 153,33

1.1.4 Proses Identifikasi

Dalam proses identifikasi dilakukan sejumlah langkah sebagai berikut :

1. Ekstraksi warna untuk memperoleh nilai RGB dari pixel-pixel yang ada pada gambar.
2. Kelompokkan nilai RGB menggunakan algoritma K-means.
3. Hitung nilai centroid dari *cluster-cluster*.
4. Nilai centroid hasil dihitung kedekatannya dengan nilai centroid dari proses pembelajaran yang sebelumnya telah disimpan ke dalam basis data dengan Euclidean *distance*. Nilai jarak yang paling minimal dipilih dengan representasi label pada data pembelajaran. Hasil proses ini adalah jumlah jenis buah yang ada pada gambar beserta namanya.

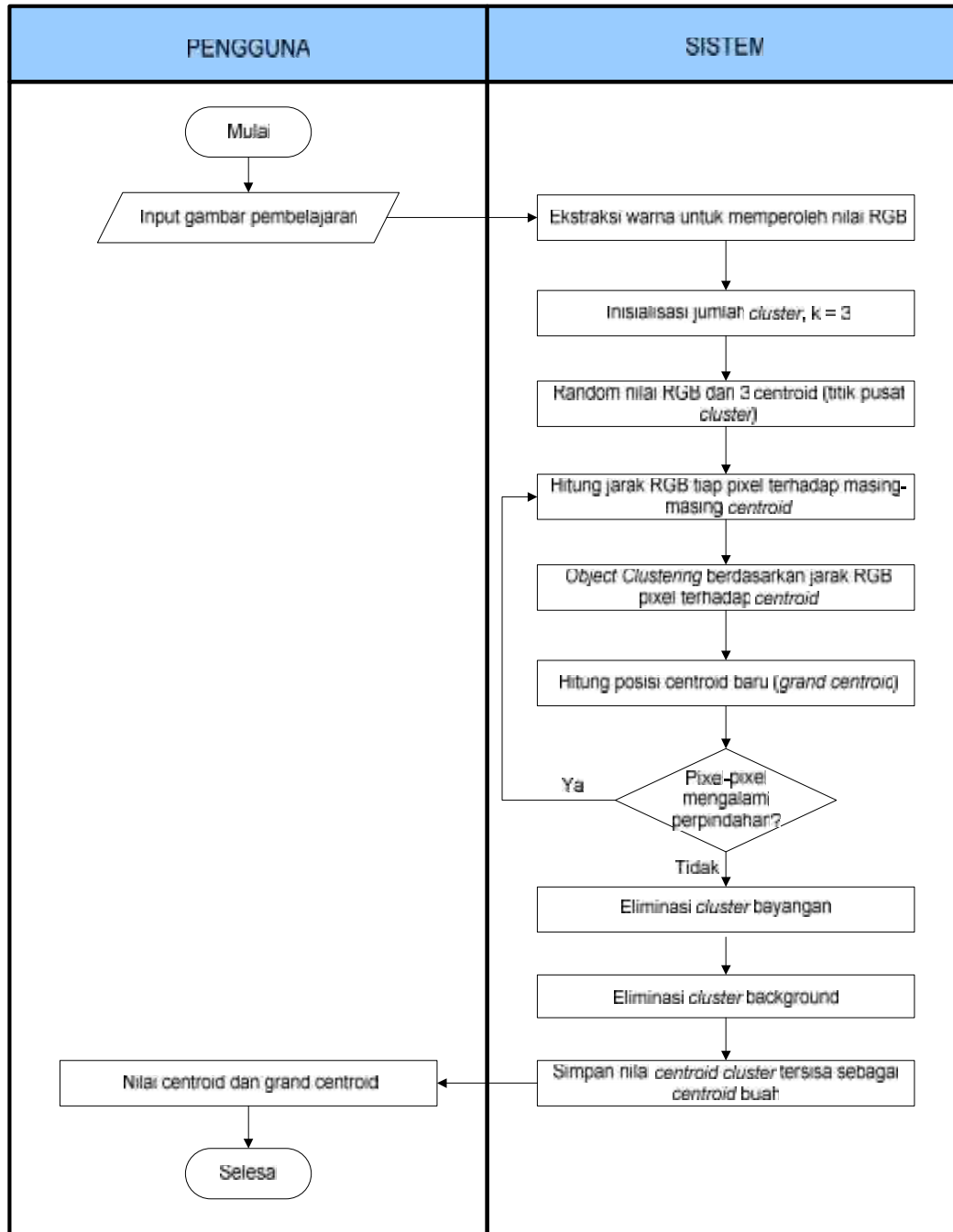
1.1.5 Deskripsi Fungsional

Analisis sistem dalam membangun Aplikasi Identifikasi Objek Buah Berdasarkan Warna dengan *Image Clustering* menggunakan Bagan Alir (*FlowChart*), *Diagram Context* dan *Data Flow Diagram* (DFD).

1.1.5.1 Flowchart

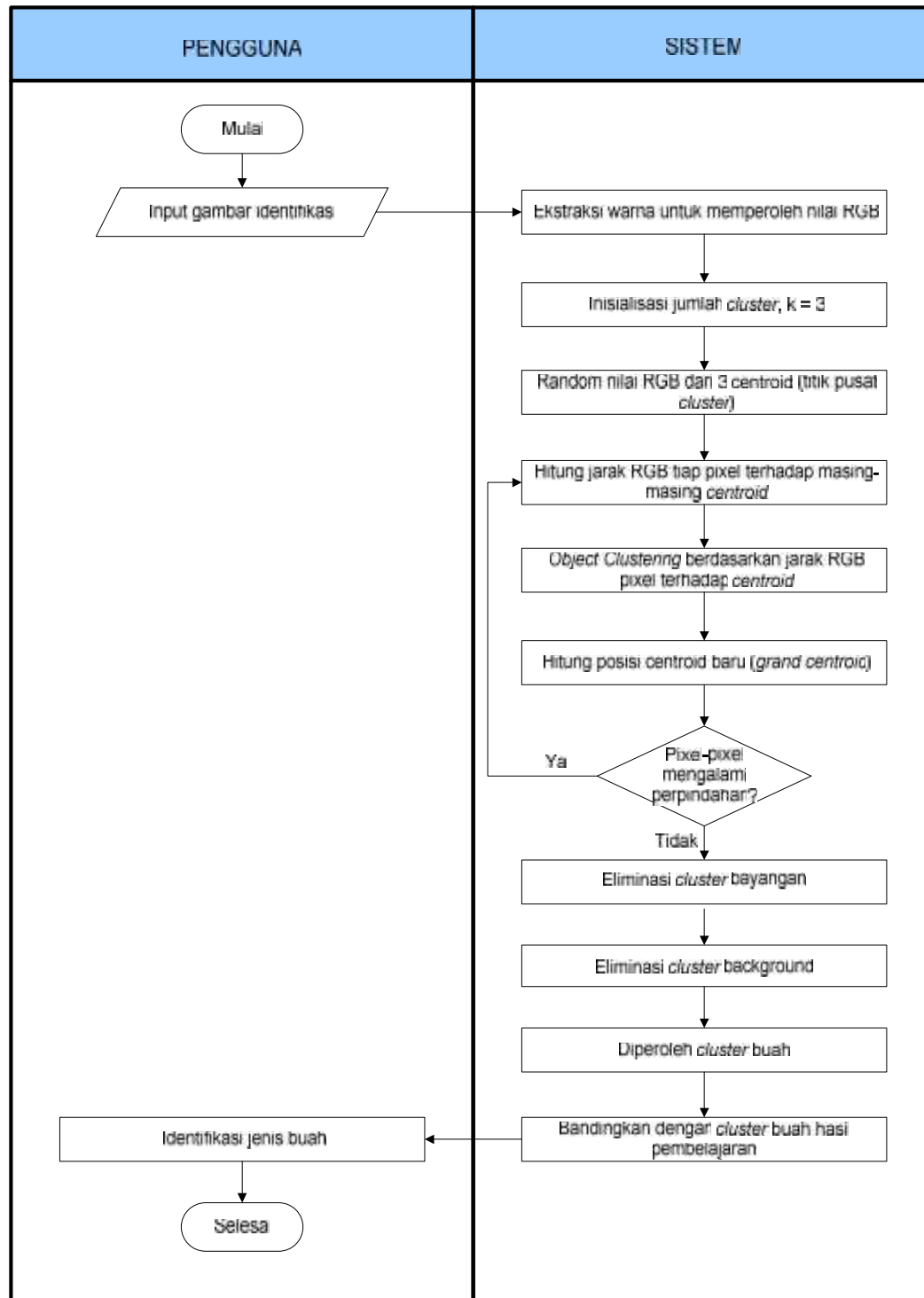
Bagan ini menjelaskan urutan-urutan dari prosedur yang ada di dalam sistem dan menunjukkan apa yang dikerjakan sistem dan pengguna.

4.1.5.3.1 Flowchart Pembelajaran



Gambar 4.2 Flowchart Pembelajaran

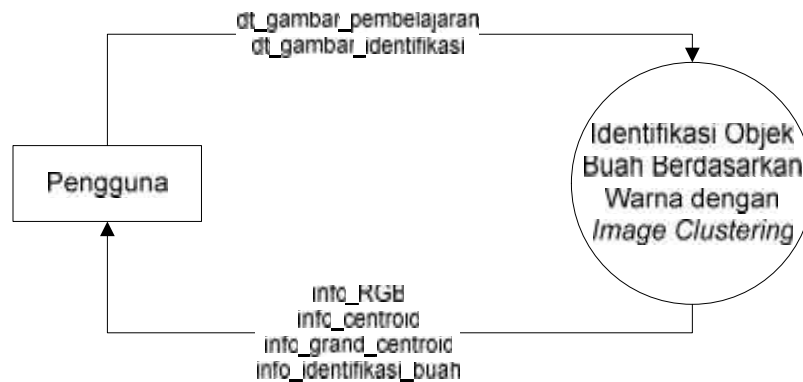
4.1.5.3.2 Flowchart Identifikasi



Gambar 4.3 Flowchart Identifikasi

1.1.5.2 Diagram Context

Diagram Context yang digunakan untuk mendeskripsikan proses aliran data Aplikasi Identifikasi Objek Buah Berdasarkan Warna dengan *Image Clustering* dapat dilihat pada gambar 4.1.



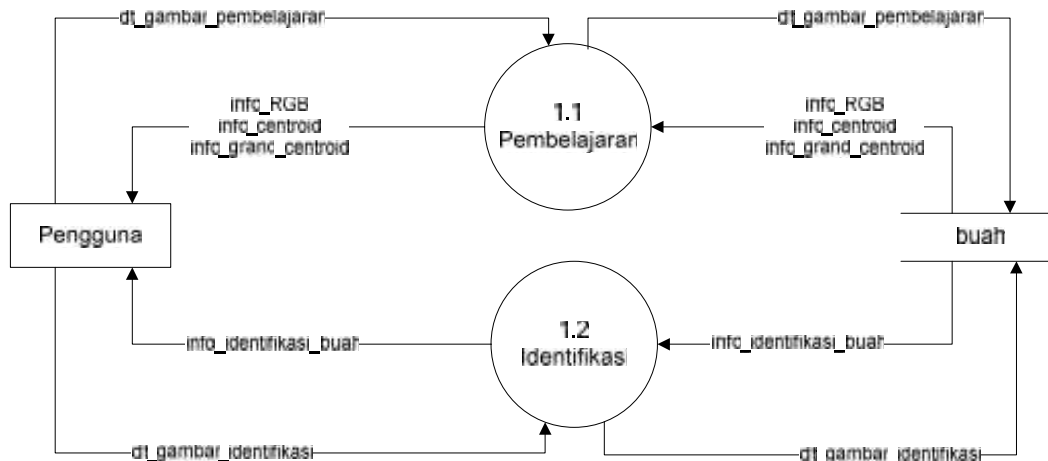
Gambar 4.4 Diagram Context

1.1.5.3 Data Flow Diagram (DFD)

Data Flow Diagram (DFD) digunakan untuk menggambarkan suatu sistem yang telah ada atau sistem baru yang akan dikembangkan secara logika.

4.1.5.3.1 DFD Level 1

Berikut ini adalah gambar DFD Level 1 aplikasi Identifikasi Objek Buah Berdasarkan Warna dengan *Image Clustering*.



Gambar 4.5

DFD Level 1 aplikasi Identifikasi Objek Buah Berdasarkan Warna dengan *Image Clustering*

Keterangan proses dan aliran data pada DFD level 1 aplikasi Identifikasi Objek Buah Berdasarkan Warna dengan *Image Clustering*.

Tabel 4.5 Proses DFD Level 1 aplikasi Identifikasi Objek Buah Berdasarkan Warna dengan *Image Clustering*

Nama	Deskripsi
Pembelajaran	Proses untuk memberikan pembelajaran kepada aplikasi mengenai nilai centroid dari masing-masing buah
Identifikasi	Proses untuk mengidentifikasi gambar buah

Tabel 4.6 Aliran Data DFD Level 1 aplikasi Identifikasi Objek Buah Berdasarkan Warna dengan *Image Clustering*

Nama	Deskripsi
dt_gambar_pembelajaran	Data gambar yang akan dicari centroid buahnya
dt_gambar_identifikasi	Data gambar yang akan diidentifikasi jenis buahnya.
info_RGB	Informasi mengenai nilai RGB pixel-pixel
info_centroid	Informasi mengenai nilai centroid buah
info_grand_centroid	Informasi mengenai nilai grand centroid buah
info_identifikasi_buah	Informasi mengenai nama buah yang teridentifikasi pada gambar.

1.2 Perancangan

Pada sub bab ini diuraikan perancangan basis data, struktur menu dan antar muka.

1.2.1 Perancangan Basis Data

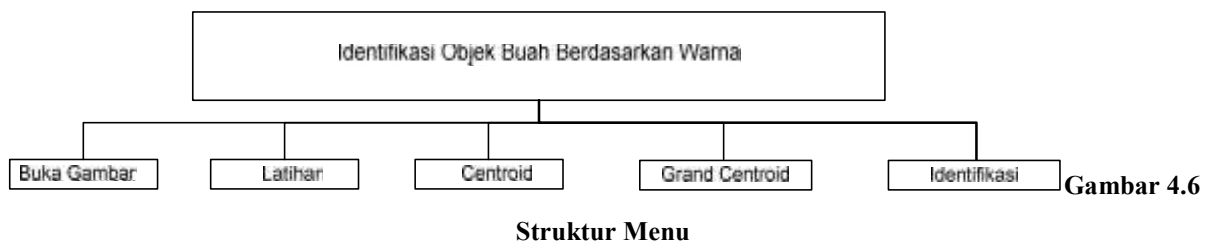
Nilai yang disimpan dalam basis data adalah nilai centroid buah yang diperoleh melalui proses pembelajaran. Basis data yang dipergunakan hanya memiliki 1 tabel yaitu **buah** dengan struktur sebagai berikut.

Tabel 4.7 Struktur Tabel Buah

Field	Tipe	Keterangan
namafile	Varchar(50)	<i>Primary Key</i>
Nama	Varchar(50)	Nama buah
R	Integer	Nilai <i>Red</i>
G	Integer	Nilai <i>Green</i>
B	Integer	Nilai <i>Blue</i>

1.2.2 Perancangan Struktur Menu

Struktur menu dari aplikasi untuk mengidentifikasi objek buah berdasarkan warna dengan metode *image clustering* meliputi:



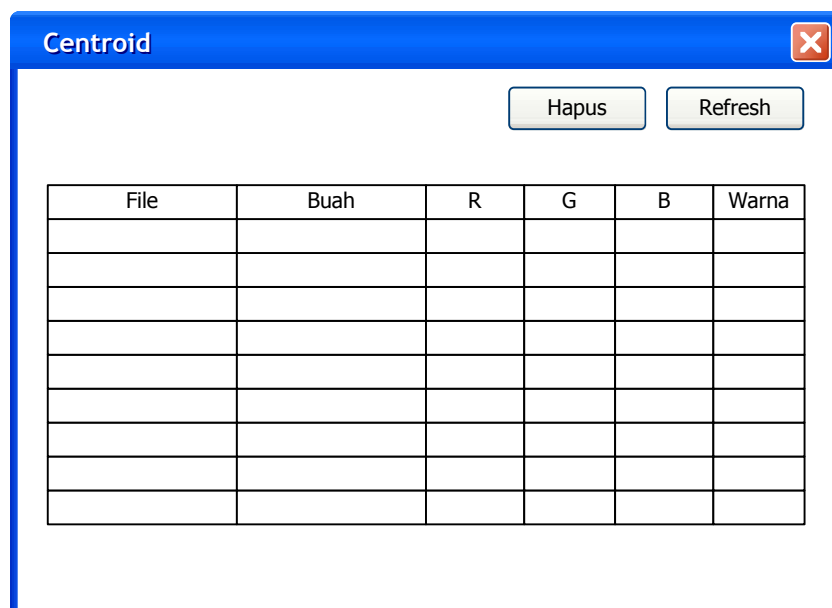
1.2.3 Perancangan Antar Muka

Sub bab ini berisikan desain antar muka dari aplikasi untuk mengidentifikasi objek buah berdasarkan warna dengan metode *image clustering*. Desain dilakukan dengan menggunakan alat bantu Microsoft Visio. Tampilan halaman utama dengan sejumlah menu (Buka Gambar, Latihan, Centroid, Grand Centroid dan Identifikasi) dapat dilihat pada gambar berikut.



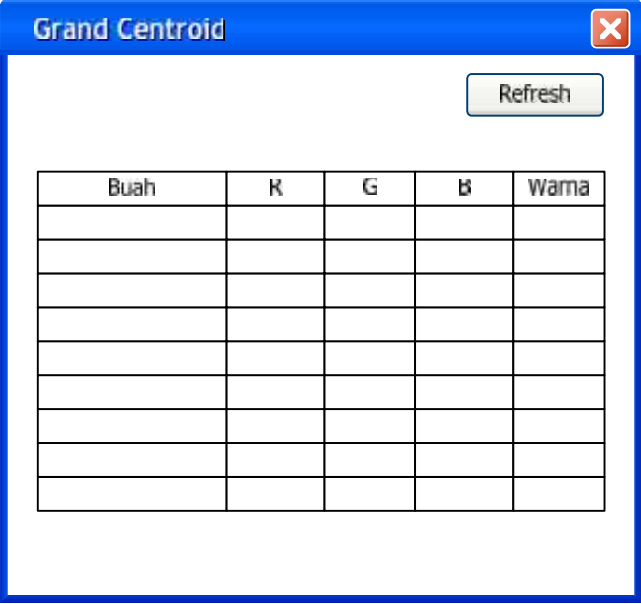
Gambar 4.7 Rancangan Tampilan Halaman Utama

Hasil pembelajaran dapat langsung diakses oleh pengguna dengan mengklik menu **Centroid** dan **Grand Centroid**. Menu **Centroid** digunakan untuk menampilkan data-data centroid dari setiap gambar buah yang telah dilatih. Data-data centroid ditampilkan dalam bentuk grid dengan kolom nama *file*, buah, nilai *red*, nilai *green*, nilai *blue*, dan representasi warnanya. Rancangan tampilan halaman centroid dapat dilihat pada gambar berikut.



Gambar 4.8 Rancangan Tampilan Centroid

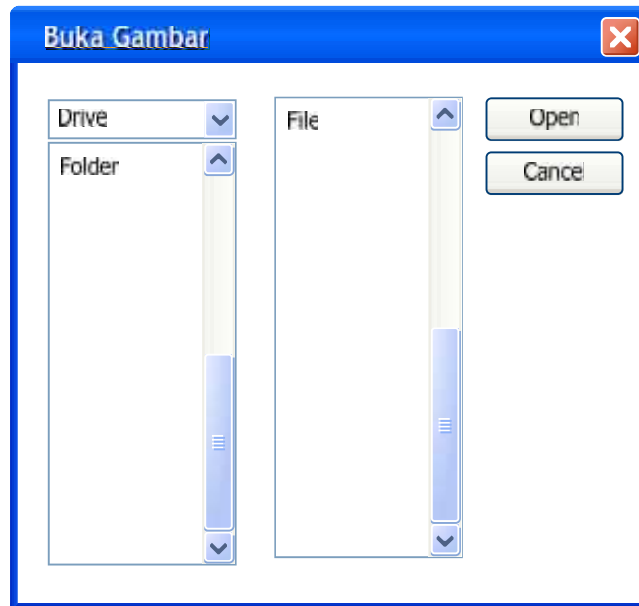
Menu **GrandCentroid** digunakan untuk menampilkan hasil perhitungan grand centroid untuk setiap label buah berdasarkan sejumlah centroid dari setiap gambar buah yang telah dilatih. Data-data grand centroid ditampilkan dalam bentuk *grid* dengan kolom nama buah, nilai *red*, nilai *green*, nilai *blue*, dan representasi warnanya. Rancangan tampilan halaman grand centroid dapat dilihat pada gambar berikut.



Buah	R	G	B	Warna

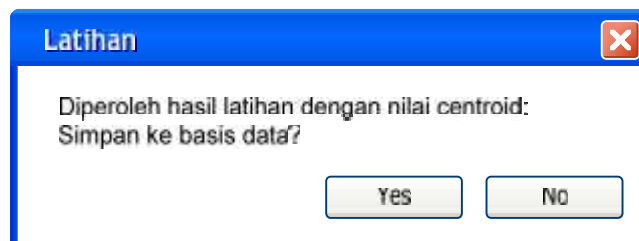
Gambar 4.9 Rancangan Tampilan Grand Centroid

Sebelum melakukan proses pembelajaran atau identifikasi, pengguna mengklik menu **Buka Gambar** untuk menampilkan gambar yang akan dipergunakan. *Browse* file gambar yang diinginkan kemudian klik tombol **Open**. Rancangan tampilan buka gambar dapat dilihat pada gambar berikut.



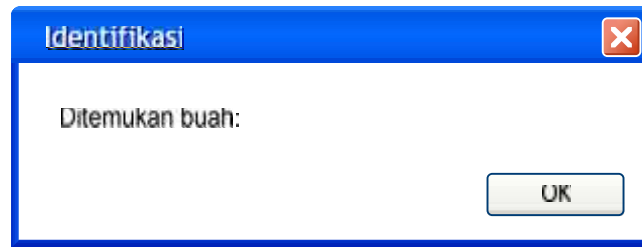
Gambar 4.10 Rancangan Tampilan Buka Gambar

Setelah file gambar ditampilkan, maka menu **Latihan** dapat digunakan untuk proses pembelajaran. Rancangan tampilan hasil latihan dapat dilihat pada gambar berikut.



Gambar 4.11 Rancangan Tampilan Pesan Latihan

Setelah file gambar ditampilkan, maka menu **Identifikasi** dapat digunakan untuk melakukan proses identifikasi dan menampilkan hasilnya. Rancangan tampilan hasil identifikasi dapat dilihat pada gambar berikut.



Gambar 4.12 Rancangan Tampilan Pesan Identifikasi

BAB V

IMPLEMENTASI DAN PENGUJIAN

1.1 Implementasi

Implementasi merupakan tahap yang dilaksanakan setelah analisa dan perancangan selesai. Pada tahap ini, pembangunan perangkat lunak dilakukan menggunakan perangkat keras dan perangkat lunak yang ditetapkan.

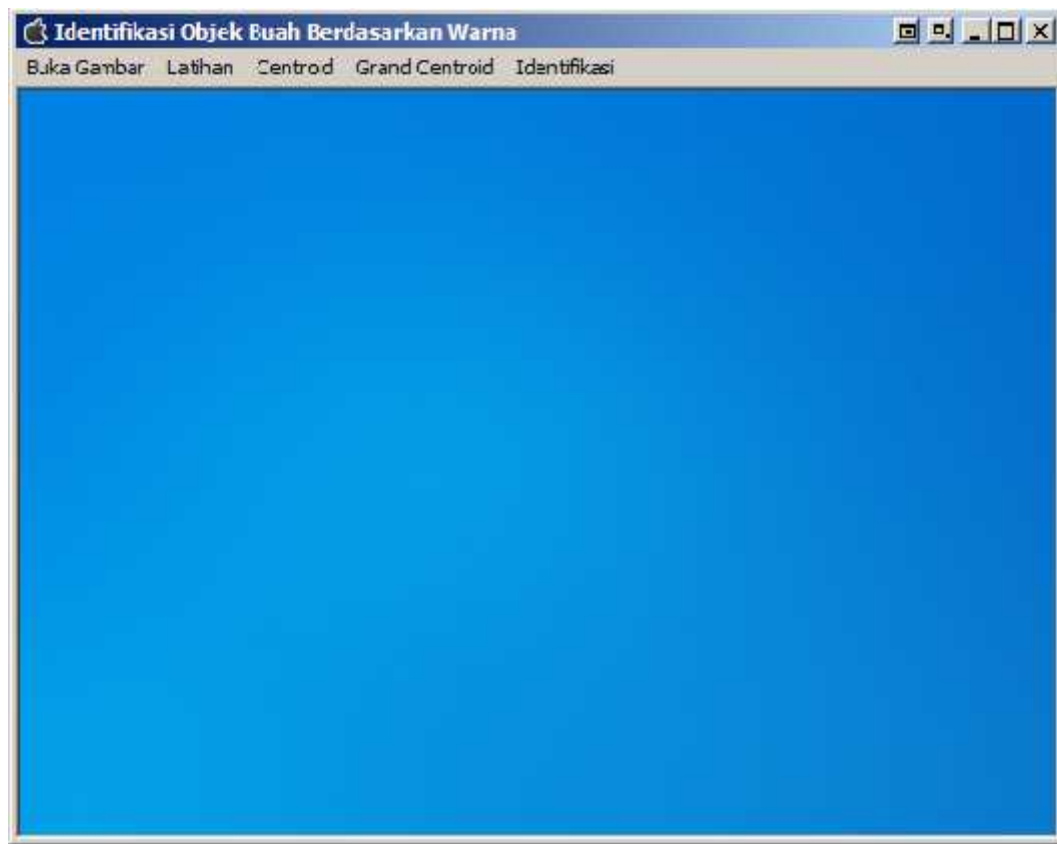
1.1.1 Lingkungan Implementasi

Lingkungan implementasi dari aplikasi untuk mengidentifikasi objek buah berdasarkan warna dengan metode *image clustering* sebagai berikut:

1. Perangkat keras berupa komputer dengan spesifikasi:
 - a. Prosesor : Intel core 2 duo
 - b. Memori : 1 GB
 - c. *Harddisk* : 80GB
2. Perangkat lunak meliputi:
 - a. Sistem Operasi : Microsoft Windows XP Professional
 - b. Pemrograman : Microsoft Visual Basic 6.0
 - c. Basis Data : MySQL 5.0

1.1.2 Hasil Implementasi

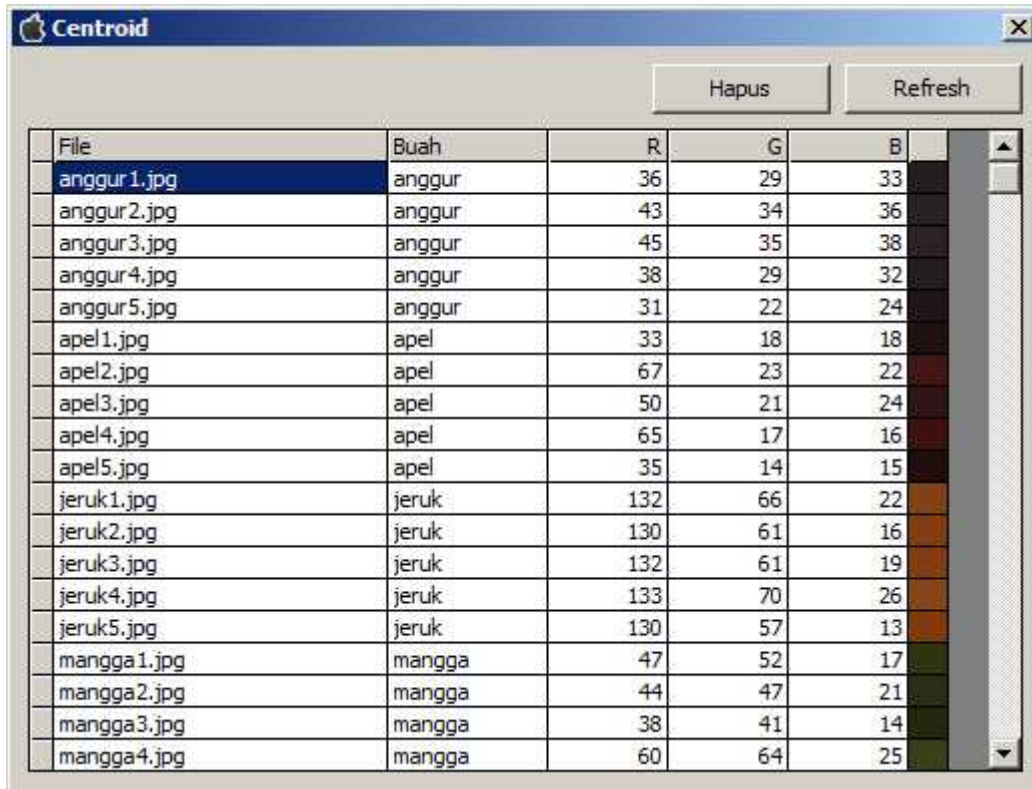
Hasil implementasi dari aplikasi untuk mengidentifikasi objek buah berdasarkan warna dengan metode *image clustering* sebagai berikut.



Gambar 5.1 Tampilan Halaman Utama

Tampilan halaman utama dari aplikasi untuk mengidentifikasi objek buah berdasarkan warna dengan metode *image clustering* dapat dilihat pada Gambar 5.1. Sejumlah menu yang tersedia meliputi: Buka Gambar, Latihan, Centroid, Grand Centroid dan Identifikasi.

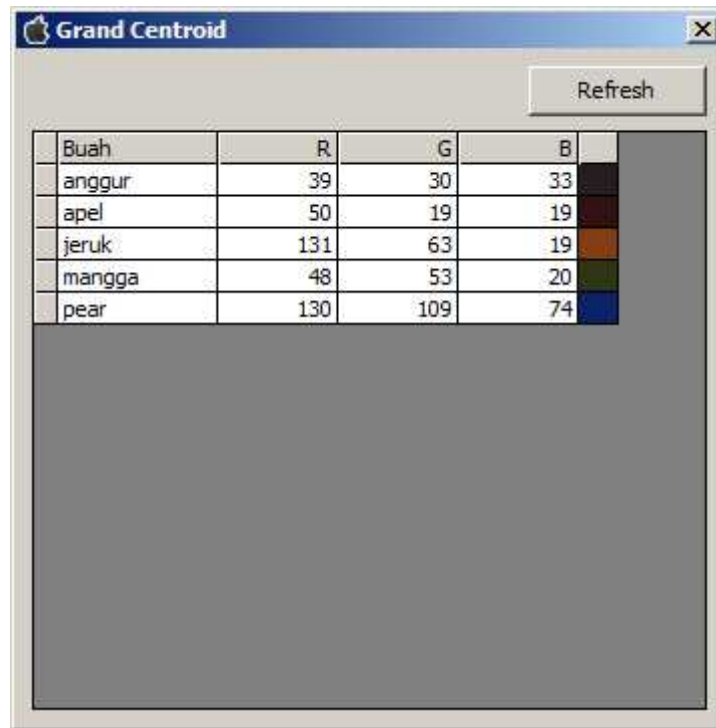
Hasil pembelajaran dapat langsung diakses oleh pengguna dengan mengklik menu **Centroid** dan **Grand Centroid**. Klik menu **Centroid** untuk menampilkan data-data centroid dari setiap gambar buah yang telah dilatih.



File	Buah	R	G	B
anggur1.jpg	anggur	36	29	33
anggur2.jpg	anggur	43	34	36
anggur3.jpg	anggur	45	35	38
anggur4.jpg	anggur	38	29	32
anggur5.jpg	anggur	31	22	24
apel1.jpg	apel	33	18	18
apel2.jpg	apel	67	23	22
apel3.jpg	apel	50	21	24
apel4.jpg	apel	65	17	16
apel5.jpg	apel	35	14	15
jeruk1.jpg	jeruk	132	66	22
jeruk2.jpg	jeruk	130	61	16
jeruk3.jpg	jeruk	132	61	19
jeruk4.jpg	jeruk	133	70	26
jeruk5.jpg	jeruk	130	57	13
mangga1.jpg	mangga	47	52	17
mangga2.jpg	mangga	44	47	21
mangga3.jpg	mangga	38	41	14
mangga4.jpg	mangga	60	64	25

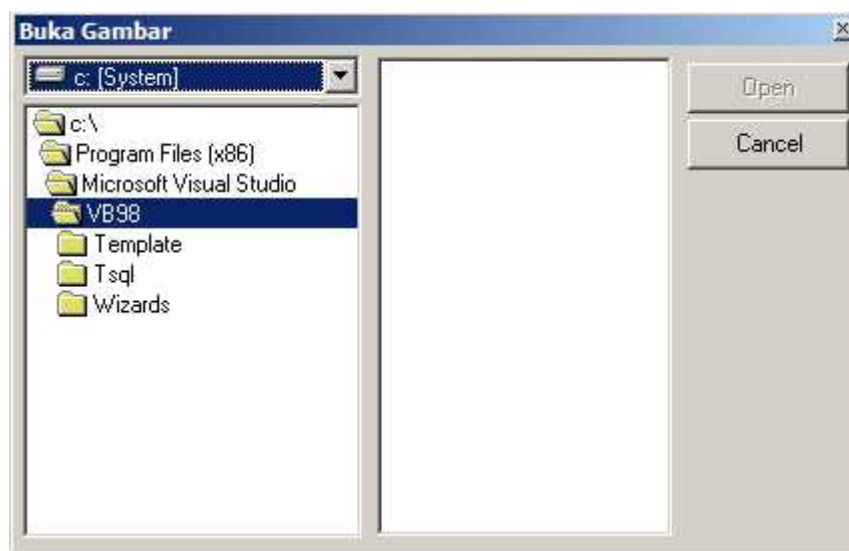
Gambar 5.2 Tampilan Centroid

Klik menu **GrandCentroid** untuk menampilkan hasil perhitungan grand centroid untuk setiap label buah berdasarkan sejumlah centroid dari setiap gambar buah yang telah dilatih.



Gambar 5.3 Tampilan Grand Centroid

Sebelum melakukan proses pembelajaran, pengguna mengklik menu **Buka Gambar** untuk menampilkan gambar yang akan dipergunakan untuk *training*. Browse file gambar yang diinginkan kemudian klik tombol **Open**.



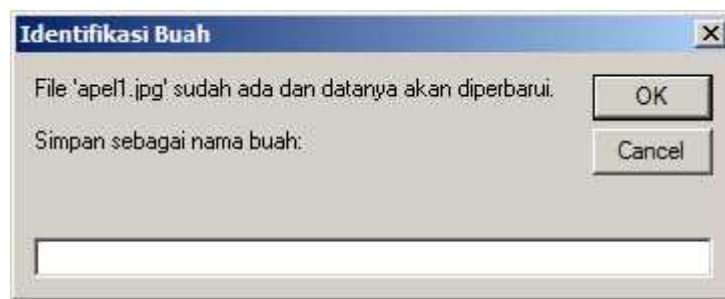
Gambar 5.4 Tampilan Buka Gambar

Selanjutnya, klik menu **Latihan** untuk memulai proses pembelajaran. Setelah proses pembelajaran selesai, ditampilkan pesan peringatan untuk menyimpan ke basis data.



Gambar 5.5 Pesan Peringatan Penyimpan Hasil Pembelajaran

Klik tombol **Yes** untuk melanjutkan ke layar masukan nama buah. Pada layar ini, masukkan nama buah yang diinginkan.



Gambar 5.6 Layar Masukan Nama Buah

Untuk melakukan proses identifikasi, pengguna mengklik menu **Buka Gambar** untuk menampilkan gambar yang akan diidentifikasi. Setelah gambar ditampilkan, klik menu **Identifikasi** untuk melakukan proses identifikasi dan menampilkan hasilnya.



Gambar 5.7 Layar Hasil Identifikasi

1.2 Pengujian

Pengujian dilakukan terhadap proses pembelajaran dan proses identifikasi. Citra masukan yang digunakan dalam aplikasi adalah citra warna (24 bit) dengan format JPEG (berekstensi file .jpg) dan berdimensi kecil, 32 x 32, sehingga dalam proses pembelajaran dan identifikasi tidak memperlama kinerja aplikasi.

Jumlah gambar yang digunakan adalah sebanyak 35, dengan rincian :

Gambar untuk pembelajaran meliputi:

1. Gambar anggur sebanyak 5 buah.
2. Gambar apel sebanyak 5 buah.
3. Gambar jeruk sebanyak 5 buah.
4. Gambar mangga sebanyak 5 buah.
5. Gambar pear sebanyak 5 buah.

Gambar untuk identifikasi sebanyak 10 buah gambar kumpulan buah berukuran 32 x 32.

1.2.1 Proses Pembelajaran

Pengujian dilakukan terhadap 5 buah gambar per masing-masing buah untuk mendapatkan nilai centroid-nya kemudian diberi label sesuai dengan nama buah yang bersangkutan. Setiap gambar buah ditampilkan satu persatu kemudian dijalankan proses pembelajaran dengan cara mengklik menu **Latihan**. Hasil pelatihan berupa centroid yang diperoleh sebagai berikut.

Tabel 5.1 Hasil Pembelajaran

File	Buah	R	G	B
anggur1.jpg	Anggur	36	29	33
anggur2.jpg	Anggur	43	34	36
anggur3.jpg	Anggur	45	35	38
anggur4.jpg	Anggur	38	29	32
anggur5.jpg	Anggur	31	22	24
apel1.jpg	Apel	33	18	18

File	Buah	R	G	B
apel2.jpg	Apel	67	23	22
apel3.jpg	Apel	50	21	24
apel4.jpg	Apel	65	17	16
apel5.jpg	Apel	35	14	15
jeruk1.jpg	Jeruk	132	66	22
jeruk2.jpg	Jeruk	130	61	16
jeruk3.jpg	Jeruk	132	61	19
jeruk4.jpg	Jeruk	133	70	26
jeruk5.jpg	Jeruk	130	57	13
mangga1.jpg	Mangga	47	52	17
mangga2.jpg	Mangga	44	47	21
mangga3.jpg	Mangga	38	41	14
mangga4.jpg	Mangga	60	64	25
mangga5.jpg	Mangga	52	59	25
pear1.jpg	Pear	135	101	48
pear2.jpg	Pear	129	122	115
pear3.jpg	Pear	134	127	121
pear4.jpg	Pear	119	92	39
pear5.jpg	Pear	133	104	49

Nilai yang dipakai untuk identifikasi adalah nilai rata-rata centroid tiap buah, disebut nilai grand centroid. Semakin banyak pembelajaran yang dilakukan pada 1 jenis buah, akan semakin baik representasi nilai grand centroid terhadap buah tersebut. Nilai RGB yang merepresentasikan masing-masing buah dapat dilihat di menu **Grand Centroid**.

1.2.2 Analisa Hasil Pembelajaran

Berikut ini adalah tabel nilai grand centroid dari masing-masing buah, dimana telah dilakukan sebanyak 5 kali percobaan masing-masingnya.

Tabel 5.2 Grand Centroid Masing-masing Buah

Buah	R	G	B
Anggur	39	30	33
Apel	50	19	19
Jeruk	131	63	19
Mangga	48	53	20
pear	130	109	74

1.2.3 Pengujian Sistem Identifikasi Buah

Berikut ini adalah pengujian sistem terhadap identifikasi buah, dimana telah dilakukan sebanyak 20 kali percobaan.

Percobaan	Buah pada Gambar	Buah Teridentifikasi	Akurasi
1	Apel, jeruk, pear, mangga	Anggur, apel, jeruk	50%
2	Pear, jeruk, mangga	Anggur, apel, jeruk, mangga	50%
3	Apel, anggur	Anggur, apel, jeruk	66%
4	Mangga, jeruk	Anggur, apel, jeruk	33%
5	Apel, anggur, pear	Anggur, apel, jeruk	66%
6	Apel, jeruk, pear	Anggur, apel, jeruk	66%
7	Apel, jeruk, pear	Anggur, apel, jeruk	66%
8	Apel, jeruk	Anggur, apel, jeruk	66%
9	Apel, jeruk, mangga	Anggur, apel, jeruk	66%
10	Apel, pear	Anggur, apel, jeruk	66%
11	Apel, jeruk, pear	Anggur, apel, jeruk, mangga	50%

12	Apel, jeruk, mangga	Anggur, apel, jeruk	66%
13	Anggur, apel, jeruk	Anggur, apel, jeruk	100%
14	Anggur, apel, jeruk	Anggur, apel, jeruk	100%
15	Apel, jeruk, mangga, pear	Anggur, apel, jeruk	50%
16	Anggur, apel, jeruk, mangga, pear	Anggur, apel, jeruk	60%
17	Anggur, apel	Anggur, apel, jeruk, mangga	50%
18	Apel, jeruk, mangga	Anggur, apel, jeruk	66%
19	Apel, pear	Anggur, apel, jeruk	33%
20	Anggur, apel, jeruk	Anggur, apel, jeruk	100%

Berdasarkan hasil pengujian sebanyak 20 kali, setelah dirata-ratakan dapat diketahui keakuratan identifikasi sebesar 63,5 %.

BAB VI

PENUTUP

1.1 Kesimpulan

Beberapa kesimpulan yang dapat diambil sebagai berikut:

1. Aplikasi yang telah dirancang bangun dapat mengidentifikasi objek buah berdasarkan ciri warna pada citra RGB (*Red, Green, dan Blue*) dengan metode *image clustering* tanpa melihat bentuk buah.
2. Proses identifikasi buah dengan algoritma K-means menggunakan metode *Valley Tracing* dalam *image clustering* mampu mengidentifikasi buah dengan keakuratan sebesar 63,5 %.

1.2 Saran

Hasil yang dicapai dari proyek akhir ini masih dapat disempurnakan untuk mencapai hasil yang lebih baik. Untuk itu diperlukan:

1. *Pre-processing* terhadap gambar yang dipergunakan dalam proses pembelajaran dan identifikasi. Hal ini dilakukan untuk mengurangi pixel-pixel yang mengganggu.
2. Penggunaan ciri lainnya dari buah, misalnya bentuk buah dan tekstur kulit buah.
3. Penggunaan gambar dengan ukuran yang lebih besar sehingga gambar terlihat dengan jelas.

DAFTAR PUSTAKA

- Acharya, T., Ray, A.K., *Image Processing Principles and Applications*, John Wiley & Sons, Inc. 2005
- Aibinu, A. M., Salami, M. J. E., Shafie, A. A., Hazali, N., Termidzi, N., *Automatic Fruits Identification System Using Hybrid Technique*, Electronic Design, Test and Applications, IEEE International Workshop, 2011
- Barakbah, A.R., Arai, K. *Determining Constraints of Moving Variance to Find Global Optimum and Make Automatic Clustering*, In. IES 2004, Politeknik Elektronika Negeri Surabaya, ITS.
- Bovik, A. *The Essential Guide to Image Processing*, Academic Press, Elsevier Inc, 2009
- Chinchuluun, R., Lee, W.S., Bhorania, J. Pardalos, P.M. *Clustering and Classification Algorithms in Food and Agricultural Applications*, Springer, 2009
- Hasniawati, H. *Image Clustering Berdasarkan Warna untuk Identifikasi Buah dengan Metode Valley Tracing*, ITS, 2007
- Meier, J. *Determining the Number of Clusters in a Data Set*, 2010. http://en.wikipedia.org/wiki/Determining_the_number_of_clusters_in_a_data_set (diakses: 1 Januari 2011)
- Ming, W.H., Hou, C.J., *Cluster analysis and visualization, Workshop on Statistics and Machine Learning*, Institute of Statistical Science, Academia Sinica, 2004.
- Morimoto, T., Takeuchi, T., Miyata, H., dan Hashimoto, Y., *Pattern Recognition of Fruit Shape Based on the Concept of Chaos and Neural*

Networks, Department of Biomechanical Systems, Ehime University,
Tarumi 3-5-7, Matsuyama 790-8566, Japan, 2000.

Nixon M., Aguado A., *Feature Extraction and Image Processing*, Academic
Press, Elsevier Inc, 2008

Putra, D. *Pengolahan Citra Digital*. Yogyakarta: Andi Offset, 2010